Answers to test: Type Theory and Coq 2010

1.

$$\lambda x : (a \to b) \to c.\, \lambda z : b.\, x\,(\lambda y : a.\, z)$$

2. (a)

$$\dfrac{\dfrac{\dfrac{\dfrac{[a \to b^x] \qquad [a^y]}{b}\, E{\to}}{c \to b}\, I[z]{\to}}{a \to c \to b}\, I[y]{\to}}{(a \to b) \to (a \to c \to b)}\, I[x]{\to}$$

(b)

$$\lambda x : a \to b.\, \lambda y : a\, \lambda z : c.\, xy$$

(c)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x : a \to b,\ y : a,\ z : c \vdash\ x : a \to b} \qquad \dfrac{}{x : a \to b,\ y : a,\ z : c \vdash\ y : a}}{x : a \to b,\ y : a,\ z : c \vdash\ xy : b}}{x : a \to b,\ y : a \vdash\ \lambda z : c.\, xy\ :\ c \to b}}{x : a \to b \vdash\ \lambda y : a.\, \lambda z : c.\, xy\ :\ a \to c \to b}}{\lambda x : a \to b.\, \lambda y : a\, \lambda z : c.\, xy\ :\ (a \to b) \to a \to c \to b}$$

3. (a)

$$\dfrac{\dfrac{\dfrac{[a^y]}{a \to a}\, I[y]{\to} \qquad [a^x]}{a}\, E{\to}}{a \to a}\, I[x]{\to}$$

(b)

$$\dfrac{[a^x]}{a \to a}\, I[x]{\to}$$

(c)

$$\lambda x : a.\, (\lambda y : a.\, y)\, x\ \to_\beta\ \lambda x : a.\, x$$

4. (a)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[Q(x)\to R(x)^v] \qquad \dfrac{\dfrac{[\forall x.\,(P(x)\to Q(x))^u]}{P(x)\to Q(x)}\,E\forall \qquad [P(x)^w]}{Q(x)}\,E{\to}}{R(x)}\,E{\to}}{P(x)\to R(x)}\,I[w]{\to}}{(Q(x)\to R(x))\to P(x)\to R(x)}\,I[v]{\to}}{\forall x.\,((Q(x)\to R(x))\to P(x)\to R(x))}\,I\forall}{(\forall x.\,(P(x)\to Q(x)))\to\forall x.\,((Q(x)\to R(x))\to P(x)\to R(x))}\,I[u]{\to}$$

(b)

$$\lambda u:(\Pi x:D.\,Px\to Qx).\,\lambda x:D.\,\lambda v:Qx\to Rx.\,\lambda w:Px.\,v(uxw)$$

5. (a)

$$\dfrac{\dfrac{\dfrac{}{\vdash *:\square} \quad \dfrac{}{\vdash *:\square}}{b:*\vdash *:\square} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash *:\square} \quad \dfrac{}{\vdash *:\square}}{b:*\vdash *:\square}}{b:*,\,a:*\vdash a:*} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash *:\square}}{b:*\vdash b:*} \quad \dfrac{\dfrac{}{\vdash *:\square} \quad \dfrac{}{\vdash *:\square}}{b:*\vdash *:\square}}{b:*,\,a:*\vdash b:*} \quad \dfrac{\dfrac{\dfrac{}{\vdash *:\square} \quad \dfrac{}{\vdash *:\square}}{b:*\vdash *:\square}}{b:*,\,a:*\vdash a:*}}{b:*,\,a:*,\,x:a\vdash b:*}}{b:*,\,a:*\vdash a\to b:*}}{b:*\vdash(\Pi a:*.\,a\to b):*}$$

(b) Yes, we can apply a function $f:\Pi a:*.\,(a\to b)$ to the object $(\Pi a:*.\,a\to b):*$, because $*$ is the type of the argument of f.

(c) The only impredicative system of the three is $\lambda 2$.

6. (a)

```
Inductive conslist : Set :=
| nil : conslist
| cons : nat -> conslist -> conslist.

Inductive snoclist : Set :=
| lin : snoclist
| snoc : snoclist -> nat -> snoclist.
```

(b)

```
forall P : snoclist -> Prop,
P lin ->
(forall l' : snoclist, P l' -> forall n : nat, P (snoc l' n)) ->
forall l : snoclist, P l
```

(c)
```
Fixpoint cons_of_snoc (l : snoclist) {struct l} : conslist :=
  match l with
  | lin => nil
  | snoc l' n => append (cons_of_snoc l') (cons n nil)
  end.
```

Or, more efficiently using tail recursion:

```
Fixpoint cons_of_snoc' (l : snoclist) (r : conslist)
    {struct l} : conslist :=
  match l with
  | lin => r
  | snoc l' n => cons_of_snoc' l' (cons n r)
  end.

Definition cons_of_snoc (l : snoclist) := cons_of_snoc' l nil.
```

7. (a) Coq notation:

```
Inductive Unary : binnat -> Prop :=
| ZERO : Unary zero
| SUCC : forall n : binnat, Unary n -> Unary (succ n).
```

Epigram notation:

$$\mathsf{data}\ \left(\frac{n : \mathtt{binnat}}{\mathtt{Unary}\, n : \mathsf{Type}}\right)\ \mathsf{where}\ \left(\frac{}{\mathtt{ZERO} : \mathtt{Unary}\, \mathtt{zero}}\right)\ \left(\frac{x : \mathtt{Unary}\, n}{\mathtt{SUCC}\, x : \mathtt{Unary}(\mathtt{succ}\, n)}\right)$$

(b) Coq notation:

```
unary : (forall n : binnat, Unary n)
```

Epigram notation:

$$\mathsf{let}\ \left(\frac{n : \mathtt{binnat}}{\mathtt{unary}\, n : \mathtt{Unary}\, n}\right)$$